

1. Advanced Features

- **User Authentication:** Add a login/signup feature with usernames and passwords. Use a simple database (e.g., SQLite) for user data.
- **Room Management:** Allow users to create, join, and leave chat rooms dynamically. Include room-level permissions (e.g., admin roles).
- **Direct Messaging:** Support private messages between users in addition to chat rooms.
- **Message History:** Store chat histories for each room and user session.
- **File Sharing:** Add the ability to send and receive files within chat rooms.

2. Robust Architecture

- **Three-Tier Architecture:** Split your application into:
 - **Client Layer:** For user interfaces and input processing.
 - **Middleware Layer:** To handle RMI communication.
 - **Server Layer:** For core logic, data storage, and thread management.
- **Thread Synchronization:** Ensure smooth multi-threading for simultaneous message delivery and room management.

3. Improved UML Diagrams

- Include detailed **sequence diagrams** for processes like login, message delivery, and room creation.
- Create a **deployment diagram** showing the interaction between clients, middleware, and server.

4. Code Quality

- Implement **logging** to monitor activities like room creation and user actions.
- Use **design patterns** such as Singleton for managing server instances and Observer for updating clients in real-time.

5. Synchronization & Concurrency

- Add thread-safe mechanisms to handle simultaneous users in the same room. For example:
 - Use **synchronized** blocks or **ReentrantLock** in Java for critical sections.
 - Handle conflicts like two users trying to create a room with the same name.

6. Scalability

- Support multiple servers for chat room distribution. Use RMI registry binding for load balancing.

7. User Interface

- If you're using a GUI (e.g., JavaFX or Swing), ensure it is intuitive and responsive.
- Add features like emoji support, message formatting, and a sidebar for active rooms/users.

8. Additional Concepts

- Implement **encryption** for secure message transmission.
- Add **error handling** for disconnected clients or failed RMI calls.

